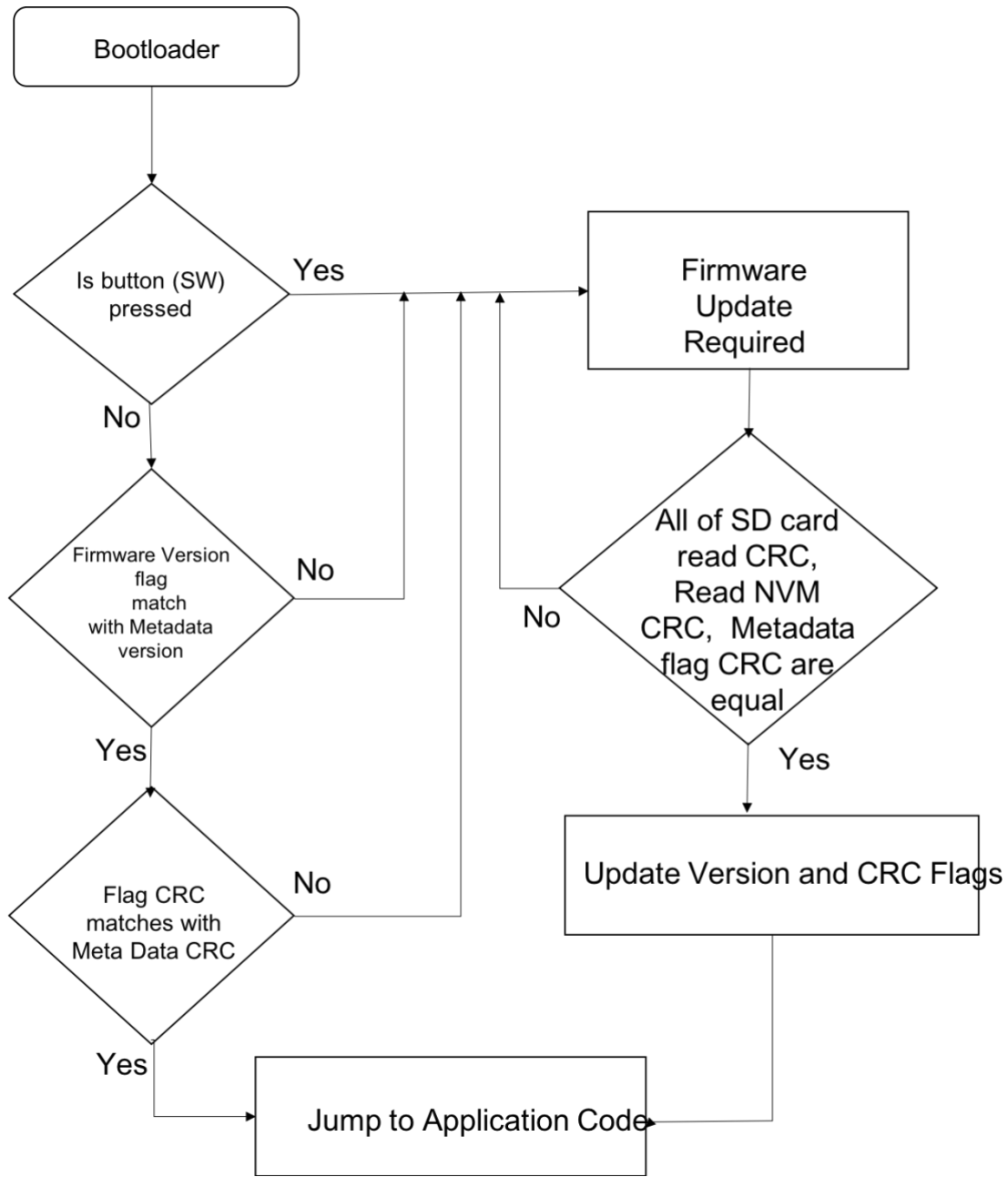


ESE 516: Assignment 6

A.) Draw out your bootloader procedure.

a. Flow chart

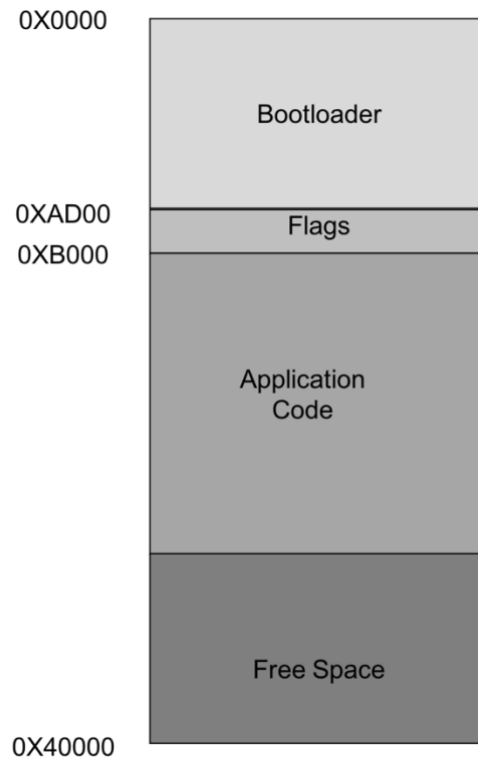


- b. In our project, firmware is updated when button is pressed, firmware version flag does not match with Metadata version, CRC flag does not match with meta data CRC and all three CRCs (SD card read, NVM read and meta data precomputed) don't match.
- c. The corrupted memory, corrupted download, power off mid download/ loading bad firmware and NVM read & write checksum (CRCs) are handled.

B.) Define your boot status & flash status flags as a struct to be saved on the SD Card or MCU NVM.

We are saving status and flags on nonvolatile memory on available space between bootloader code and application code. The struct having version flag and checksum CRC 32 flag are stored using struct. We are writing and reading struct on NVM.

C) Define your partition tables for MCU Non-volatile Memory



The screen shots below show map for the project.

The program start address is 0x000 where bootloader code begins

Memory 4												
Memory:		base FLASH										
0xFFFFFD0	??	??	??	??	??	??	??	??	??	??	??	??
0xFFFFFE0	??	??	??	??	??	??	??	??	??	??	??	??
0xFFFFF00	??	??	??	??	??	??	??	??	??	??	??	??
0x0000000	98	2c	00	20	9d	34	00	00	99	34	00	00
0x0000010	00	00	00	00	00	00	00	00	00	00	00	00
0x0000020	00	00	00	00	00	00	00	00	00	00	99	34
0x0000030	00	00	00	00	00	00	00	00	99	34	00	00
0x0000040	99	34	00	00	99	34	00	00	99	34	00	00
0x0000050	99	34	00	00	99	34	00	00	99	34	00	00
0x0000060	99	34	00	00	8d	1f	00	00	9d	1f	00	ad
0x0000070	bd	1f	00	00	cd	1f	00	00	dd	1f	00	99
0x0000080	99	34	00	00	99	34	00	00	99	34	00	00
0x0000090	99	34	00	00	00	00	00	00	00	00	99	34
0x00000A0	99	34	00	00	99	34	00	00	99	34	00	00
0x00000B0	00	00	00	00	10	b5	06	4c	23	78	00	2b
0x00000C0	00	2b	02	d0	04	48	00	e0	00	bf	01	23
0x00000D0	84	00	00	20	00	00	00	00	04	6e	00	08
0x00000E0	00	2b	03	d0	07	49	08	48	00	e0	00	bf
0x00000F0	00	2b	00	d1	10	bd	06	4b	00	2b	fb	d0
0x0000100	00	00	00	00	88	00	00	20	04	6e	00	04
0x0000110	00	00	00	00	f0	b5	c6	46	00	b5	03	23
0x0000120	13	68	db	43	0c	00	03	29	02	d9	44	42
0x0000130	00	2c	16	d0	09	1b	3e	1d	3d	68	e0	00
0x0000140	c5	40	6b	40	e0	00	19	d0	01	24	27	4d
0x0000150	01	38	00	28	04	d0	1c	42	f9	d0	5b	08
0x0000160	37	00	03	20	0c	00	84	43	1e	d0	08	1b
0x0000170	88	46	3e	00	0d	00	01	20	1b	4c	0b	e0
0x0000180	5b	08	01	39	00	29	04	d0	18	42	f9	d0
0x0000190	f7	e7	04	36	01	3d	03	d3	31	68	4b	40
0x00001A0	41	46	89	00	7f	18	61	46	00	29	17	d0
0x00001B0	40	42	20	30	01	25	6d	42	c5	40	28	00
0x00001C0	c9	00	0b	d0	01	20	08	4c	03	e0	5b	08
0x00001D0	04	d0	18	42	f9	d0	5b	08	63	40	f7	db
0x00001E0	00	20	04	bc	90	46	f0	bd	20	83	b8	ed
0x00001F0	01	ac	01	25	65	70	00	27	a7	70	25	70
0x0000200	06	4e	b0	47	80	22	12	04	05	4b	9a	61
0x0000210	21	00	37	20	b0	47	03	b0	f0	bd	c0	46

The application code address starts at 0xB000 on NVM

Memory 4												
Memory:		base FLASH										
0x000AFD0	ff	ff	ff	ff	ff	ff	ff	ff	ff	ff	ff	ff
0x000AFE0	ff	ff	ff	ff	ff	ff	ff	ff	ff	ff	ff	ff
0x000AFF0	ff	ff	ff	ff	ff	ff	ff	ff	ff	ff	ff	ff
0x000B000	e8	25	00	20	95	e0	00	00	91	e0	00	00
0x000B010	00	00	00	00	00	00	00	00	00	00	00	00
0x000B020	00	00	00	00	00	00	00	00	00	00	91	e0
0x000B030	00	00	00	00	00	00	00	00	91	e0	00	00
0x000B040	91	e0	00	00	91	e0	00	00	91	e0	00	00
0x000B050	91	e0	00	00	91	e0	00	00	91	e0	00	00
0x000B060	91	e0	00	00	d1	cb	00	00	e1	cb	00	f1
0x000B070	01	cc	00	00	11	cc	00	00	21	cc	00	91
0x000B080	91	e0	00	00	91	e0	00	00	91	e0	00	00
0x000B090	91	e0	00	00	00	00	00	00	00	00	00	91
0x000B0A0	91	e0	00	00	91	e0	00	00	91	e0	00	00
0x000B0B0	00	00	00	00	10	b5	06	4c	23	78	00	2b
0x000B0C0	00	2b	02	d0	04	48	00	e0	00	bf	01	23
0x000B0D0	68	00	00	20	00	00	00	00	30	0b	01	00
0x000B0E0	00	2b	03	d0	07	49	08	48	00	e0	00	bf
0x000B0F0	00	2b	00	d1	10	bd	06	4b	00	2b	fb	d0
0x000B100	00	00	00	00	6c	00	00	20	30	0b	01	00
0x000B110	00	00	00	00	f0	b5	83	b0	01	ac	01	25
0x000B120	a7	70	25	70	21	00	17	20	06	4e	b0	47
0x000B130	05	4b	9a	61	27	70	65	70	21	00	37	20
0x000B140	f0	bd	c0	46	f1	c5	00	00	00	44	00	41
0x000B150	03	7c	00	2b	05	d0	83	68	58	1c	e1	68
0x000B160	a1	60	63	68	58	1c	e1	68	04	4b	98	47

The flags are stored on NVM in a struct at address 0XAD00

Memory 4																
Memory:		base FLASH														
0x0000ACD0		ff	ff	ff	ff	ff	ff	ff	ff	ff	ff	ff	ff	ff	ff	ff
0x0000ACE0		ff	ff	ff	ff	ff	ff	ff	ff	ff	ff	ff	ff	ff	ff	ff
0x0000ACF0		ff	ff	ff	ff	ff	ff	ff	ff	ff	ff	ff	ff	ff	ff	ff
0x0000AD00		33	0d	0a	00	00	00	00	00	00	00	00	00	00	00	00
0x0000AD10		00	00	00	00	00	00	00	00	00	00	00	00	00	00	00
0x0000AD20		32	63	39	39	37	38	32	31	00	00	00	00	00	00	00
0x0000AD30		00	00	00	00	00	00	00	00	00	00	00	00	00	00	00
0x0000AD40		ff	ff	ff	ff	ff	ff	ff	ff	ff	ff	ff	ff	ff	ff	ff
0x0000AD50		ff	ff	ff	ff	ff	ff	ff	ff	ff	ff	ff	ff	ff	ff	ff
0x0000AD60		ff	ff	ff	ff	ff	ff	ff	ff	ff	ff	ff	ff	ff	ff	ff

D) Create an Atmel Studio solution with two projects, Bootloader & Application Code.

The project having bootloader and application code is attached in Radince.zip file on A6 google drive. All the checks and functions to read from SD card, read from SD card and writing on NVM are implemented and explained in the code.

The NVM row size is 256 B and page size is 64 B. There are 4 pages in each row. The number of rows required are calculated as (size of application object / Row size)

The outputs and explanations are printed on using serial write console on serial terminal. It prints necessary updates in serial consol.